IN THE SPECIFICATION:

Please amend the specification as follows.

Please replace the section on page 1, lines 4-8 with the following amended section:

CROSS-REFERENCE TO RELATED APPLICATIONS

The present application is related to U.S. Patent Application No. ~~(not yet assigned)~~ 09/697,730, ~~(Attorney Docket No. SUN1P412),~~ filed on October 25, 2000, entitled, "HIGH PERFORMANCE TRANSMISSION LINK AND INTERCONNECT", which is incorporated herein by reference.

Also, please replace the paragraph on page 3, line 23 – page 4, line 10 with the following amended paragraph:

In another aspect of the present invention, a method of transmitting data over a network having multiple nodes and links when a link has failed is described. A data packet is received at a first node having a failed link where the data packet is scheduled to use the failed link. Data goes into the transmission buffer of the failed link upon exiting the transmit buffer. It is converted and sent to a failover storage area. The link is chosen when the packet is pushed to the failover storage area. The data packet is routed to a failover storage area. When the packet is pushed to the failover storage area, an ~~alternatate~~ alternate link is selected for the data packet and the data packet is routed to a transmitter associated with the alternative link. The data packet is then transmitted to a receiver for the alternative link at a destination or multihop node. This allows the data packet to reach an intended destination by modifying the first node at a hardware level and without any software intervention.

Furthermore, please replace the paragraph on page 10, line 22 – page 11, line 19 with the following amended paragraph:

At step 302 Node 0 detects a failure in Link A. As is known in the field, this can be done by examining the transmission error rate on a link and comparing it to a threshold number of errors. As mentioned above, a link is a bidirectional path implemented using two unidirectional physical segments. When one of the segments fails, both segments on the bidirectional link go into failover mode. One end (referred to as the near-end) experiences an excessive transmission error rate causing it to enter failover mode. In this example, the near-end is Node 0 and, specifically, the transmitter for Link A. The near-end will attempt to signal the far-end of this, using an administrative packet sent on an oppositely-directed link (*i.e.*, the link that is connected to the transmitter associated with the receiver) before shutdown, where the administrative packet has an *in_failover* bit or an equivalent bit set. The far-end is the receiver for Link A in Node 1. This is shown at step 304. If the far-end receives this failover notification, it will also go into failover mode. The far-end receiver may not receive this advisory packet because the link may not be reliable (administration packets do not have retransmission features). If not, the ensuing transmission errors resulting from the near-end link shutdown (*i.e.*, Node 0 turns off its clock) will cause the far-end to go into failover mode. More specifically, in certain networks, after sending [[16]] sixteen administrative packets with an *in_failover* bit set, the hardware in Node 0 will turn off the link clock on the failed segment. This process insures that an administrative packet will have been sent with an *in_failover* bit set on the failed link. If this packet is not received by the far-end, the far-end (*i.e.*, Node 1) will enter failover mode due to clocking errors it detects as a result of the clock being turned off.